120

AF/2134

Practitioner's Docket No. NAI1P250/00.024.01 *PATENT*

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Cheuk W. Ko et al.

Application No.: 09/593,280 Group No.: 2134
Filed: 06/30/2000 Examiner: Heneghan, M.
For: METHOD AND APPARATUS FOR CONTENT-BASED INTRUSION DETECTION USING AN AGILE KERNEL-BASED AUDITOR

Mail Stop Appeal Briefs – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF
(PATENT APPLICATION--37 C.F.R. § 1.192)

1. Transmitted herewith, in triplicate, is the APPEAL BRIEF in this application, with respect to the Notice of Appeal filed on April 8, 2004.

2. STATUS OF APPLICANT

This application is on behalf of other than a small entity.

CERTIFICATION UNDER 37 C.F.R. §§ 1.8(a) and 1.10*

*(When using Express Mail, the Express Mail label number is **mandatory**; Express Mail certification is optional.)*

I hereby certify that, on the date shown below, this correspondence is being:

MAILING

☒ deposited with the United States Postal Service in an envelope addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

37 C.F.R. § 1.8(a)
_ with sufficient postage as first class mail.

37 C.F.R. § 1.10*
_ as "Express Mail Post Office to Addressee"
Mailing Label No. _____ **(mandatory)**

TRANSMISSION

_ facsimile transmitted to the Patent and Trademark Office, (703) _____ - _____.

Melissa D Orvis
Signature

Date: 4/30/04

Melissa D. Orvis
(type or print name of person certifying)

** Only the date of filing (' 1.6) will be the date used in a patent term adjustment calculation, although the date on any certificate of mailing or transmission under ' 1.8 continues to be taken into account in determining timeliness. See ' 1.703(f). Consider "Express Mail Post Office to Addressee" (' 1.10) or facsimile transmission (' 1.6(d)) for the reply to be accorded the earliest possible filing date for patent term adjustment calculations.*

3. FEE FOR FILING APPEAL BRIEF

Pursuant to 37 C.F.R. § 1.17(c), the fee for filing the Appeal Brief is:

other than a small entity $330.00

Appeal Brief fee due $330.00

4. EXTENSION OF TERM

The proceedings herein are for a patent application and the provisions of 37 C.F.R. § 1.136 apply.

Applicant believes that no extension of term is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

5. TOTAL FEE DUE

The total fee due is:

Appeal brief fee	$330.00
Extension fee (if any)	$0.00

TOTAL FEE DUE $330.00

6. FEE PAYMENT

Attached is a check in the amount of $330.00.

A duplicate of this transmittal is attached.

7. FEE DEFICIENCY

If any additional extension and/or fee is required, and if any additional fee for claims is required, charge Deposit Account No. 50-1351 (Order No. NAI1P250).

Signature of Practitioner

Reg. No.: 41,429
Tel. No.: 408-971-2573
Customer No.: 28875

Kevin J. Zilka
Silicon Valley IP Group, PC
P.O. Box 721120
San Jose, CA 95172-1120
USA

#12

Practitioner's Docket No. NAI1P250/00.024.01
PATENT 103

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	)	
	)	
Cheuk W. Ko	)	Art Unit: 2134
	)	
Application No. 09/593,280	)	Ex.: Heneghan
Filed: 6/13/00	)	
	)	
For: METHOD AND APPARATUS FOR CONTENT-BASED INTRUSION DETECTION USING AN AGILE KERNEL-BASED AUDITOR	)	

Commissioner for Patents
Alexandria, VA 22313-1450

ATTENTION: Board of Patent Appeals and Interferences

APPELLANT'S BRIEF (37 C.F.R. § 1.192)

This brief is in furtherance of the Notice of Appeal, filed in this case on April 8, 2004.

CERTIFICATION UNDER 37 C.F.R. ' ' 1.8(a) and 1.10*
(When using Express Mail, the Express Mail label number is ***mandatory;***
Express Mail certification is optional.)

I hereby certify that, on the date shown below, this correspondence is being:

MAILING

X deposited with the United States Postal Service in an envelope addressed to the Commissioner for Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

37 C.F.R. § 1.8(a)
_ with sufficient postage as first class mail.

37 C.F.R. § 1.10*
_ as "Express Mail Post Office to Addressee"
Mailing Label No. _____ **(mandatory)**

TRANSMISSION

_ facsimile transmitted to the Patent and Trademark Office, (703) 872-9306.

Date: 4/30/04

Melissa D. Orvis
Signature

Melissa D. Orvis
(type or print name of person certifying)

** Only the date of filing (' 1.6) will be the date used in a patent term adjustment calculation, although the date on any certificate of mailing or transmission under ' 1.8 continues to be taken into account in determining timeliness. See ' 1.703(f). Consider "Express Mail Post Office to Addressee" (' 1.10) or facsimile transmission (' 1.6(d)) for the reply to be accorded the earliest possible filing date for patent term adjustment calculations.*

The fees required under § 1.17, and any required petition for extension of time for filing this brief and fees therefor, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief is transmitted in triplicate. (37 C.F.R. § 1.192(a))

This brief contains these items under the following headings, and in the order set forth below (37 C.F.R. § 1.192(c)):

I REAL PARTY IN INTEREST
II RELATED APPEALS AND INTERFERENCES
III STATUS OF CLAIMS
IV STATUS OF AMENDMENTS
V SUMMARY OF INVENTION
VI ISSUES
VII GROUPING OF CLAIMS
VIII ARGUMENTS
APPENDIX OF CLAIMS INVOLVED IN THE APPEAL

The final page of this brief bears the practitioner's signature.

I REAL PARTY IN INTEREST (37 C.F.R. § 1.192(c)(1))

The real party in interest in this appeal is Networks Associates Technology, Inc.

II RELATED APPEALS AND INTERFERENCES (37 C.F.R. § 1.192(c)(2))

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no other such appeals or interferences.

III STATUS OF CLAIMS (37 C.F.R. § 1.192(c)(3))

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-27.

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims withdrawn from consideration but not canceled: None
2. Claims pending: 1-27
3. Claims allowed: None
4. Claims rejected: 1-27

C. CLAIMS ON APPEAL

The claims on appeal are: 1-27

IV STATUS OF AMENDMENTS (37 C.F.R. § 1.192(c)(4))

As to the status of any amendment filed subsequent to final rejection, there are no such amendments after final.

V SUMMARY OF INVENTION (37 C.F.R. § 1.192(c)(5))

A method and computer program product are provided for content-based intrusion detection for a computer system by using an agile kernel-based auditing system. As set forth in Figure 4 and the accompanying description, such auditing system operates by receiving an audit specification that specifies target attributes to be recorded during an auditing process. The audit specification also specifies an auditing criterion that triggers recording of the target attributes. Upon receiving the audit specification, the auditing system is configured to record the target attributes during system calls whenever the auditing criterion is satisfied. Next, an application program is monitored by the auditing system to produce an audit log containing the recorded target attributes.

Next, as set forth in operation 412 of Figure 4 and the accompanying description, this audit log is then examined in order to detect patterns for intrusion detection purposes. In one embodiment of the present invention, configuring the auditing system involves compiling the audit specification to produce a kernel module, and then loading the kernel module into a kernel of an operating system. It also involves linking code from within the kernel module into system calls within the operating system.

As set forth in operation 416 of Figure 4 and the accompanying description, in response to detecting an event during the auditing process, the system dynamically adjusts the auditing system to change the auditing criterion and/or the target attributes for subsequent operation of the auditing system.

VI ISSUES (37 C.F.R. § 1.192(c)(6))

Issue # 1: The Examiner has rejected Claims 1-3, 7-12, 16-21, and 25-27 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No.: 5,557,742 to Smaha et al., in view of U.S. Patent No.: 5,513,317 to Borchardt et al.

Issue # 2: The Examiner has further rejected Claims 4, 6, 13, 15, 22, and 24 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,557,742 to Smaha et al., in view of U.S. Patent No.: 5,513,317 to Borchardt et al., further in view of U.S. Patent No. 6,584,508 to Epstein et al., and further in view of Kernighan et al., "The UNIX Programming Environment," 1984.

Issue # 3: The Examiner has still further rejected Claims 5, 14, and 23 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,557,742 to Smaha et al., in view of U.S. Patent No.: 5,513,317 to Borchardt et al., and further in view of U.S. Patent No. 5,623,601 to Vu.

VII GROUPING OF CLAIMS (37 C.F.R. § 1.192(c)(7))

The claims of the above groups do not stand or fall together. Following is the grouping of claims. In the following section, appellant explains why the claims of each group are believed to be separately patentable.

Issue # 1: Grouping of Claims –
Group #1: 1-2, 7-12, 17-20, and 26-27;
Group #2: 3, 13, and 21; and
Group #3: 7, 16, and 25.

Issue # 2: Grouping of Claims –

Group #1: Claims 4, 13, and 22; and
Group #2: Claims 6, 15, and 24.

Issue # 3: Grouping of Claims –
Group #1: Claims 5, 14, and 23.

VIII ARGUMENTS (37 C.F.R. § 1.192(c)(8))

Issue #1:

The Examiner has rejected Claims 1-3, 7-12, 16-21, and 25-27 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No.: 5,557,742 to Smaha et al., in view of U.S. Patent No.: 5,513,317 to Borchardt et al.

Group #1: Claims 1-2, 7-12, 17-20, and 26-27

With respect to Claims 1-2, 7-12, 17-20, and 26-27; it appears that the Examiner continues to rely on Smaha to make a prior art showing of appellant's claimed:

"receiving an audit specification;

wherein the audit specification specifies at least one target attribute to be recorded from a set of possible target attributes during an auditing process by the auditing system;

wherein the audit specification also specifies at least one auditing criterion that triggers recording of the at least one target attribute during the auditing process;

configuring the auditing system to record the at least one target attribute in response to detecting the at least one auditing criterion."

Specifically, the Examiner has now equated appellant's claimed "audit specification" to Smaha's "selected misuses," wherein a misuse is a "target attribute."

This is simply incorrect. Appellant's claimed "audit specification" is not a misuse, but rather a specification that is used to configure an auditing system for creating an audit log that is, in turn, examined to detect patterns for intrusion detection purposes, as defined by appellant's claims. Thus, it is appellant's claimed "patterns" that are most analogous to Smaha's "selected misuses" and associated signature data structures, not appellant's claimed "audit specification."

It appears that the Examiner is attempting to arbitrarily map various terms in appellant's claims with those in the Smaha reference. Specifically, it appears that the Examiner is relying on Smaha's "selected misuses" (and associated signatures) to meet both appellant's claimed "audit specification" and "patterns." However, such attempt clearly fails, since the remaining functionality of appellant's claims is simply not met.

For example, simply nowhere in Smaha are the "selected misuses" (which are allegedly equivalent to appellant's claimed "audit specification," per the Examiner) used to configure an audit system to produce an audit log which is, in turn, "examin[ed] ... to detect patterns for intrusion detection purposes." While Smaha's "selected misuses" may be used to detect patterns for intrusion detection purposes, they are simply *not* used to configure an audit system to produce an

audit log for recording purposes, as claimed. It appears that the Examiner is improperly attempting to use a single entity (i.e. "selected misuses") in Smaha to meet two entities (i.e. "audit specification" and "patterns") in appellant's claims. This obviously fails, especially since the related functionality is not met by Smaha.

Again, the primary reason that Smaha fails is due to the fact that it does not disclose the crux of appellant's claimed invention that is embodied in the claims, nor the problem that it solves. As indicated on page 11, first paragraph of the originally file specification: by selectively recording target attributes, the present invention can reduce the amount of data that is recorded during the auditing process. This makes it practical to record data that is read or written during system calls without overwhelming the storage capacity, processing power and/or data transfer bandwidth of a computer system.

Appellant's claimed invention thus limits the amount of data that is even subjected to pattern (i.e. "misuse, etc.) detection.

In the latest action mailed March 22, 2004, the Examiner responds to the above arguments by stating that "a recitation of the intended use of the claimed invention must result in a structural difference between the claimed invention and the prior art in order to patentably distinguish the claimed invention from the prior art. If the prior art structure is capable of performing the intended use, then it meets the claim. ..." The Examiner goes on to state that appellant's "audit specification" is read as being a listing of attributes that are to be monitored. The Examiner further asserts that no further definition above and beyond this is made in appellant's specification.

In response to such latest arguments, appellant again points to the above arguments where it is clearly explained that appellant has claimed an "audit specification" that specifically configures an audit system to produce an audit log which is, in turn, is "examin[ed] ... to detect patterns for intrusion detection purposes." Smaha's "misuses" clearly do not meet appellant's functionality that is claimed in conjunction with the "audit specification." Thus, the prior art is not capable of performing the intended use, as suggested by the Examiner. More importantly, the functional limitations that surround the claimed "audit specification" have clearly not been met.

Moreover, it appears that the Examiner has not adequately addressed appellant's remaining arguments.

Specifically, the Examiner relies on the following excerpt from Borchardt et al. to make a prior art showing of appellant's claimed "size of the audit log [being] reduced when the auditing system is run prior to the examination for detection of the patterns."

> "First, the program is executed while the trace facility is active (Step 104). However, rather than immediately supply text representation of the filtered entries to the programmer, the trace facility gathers all historical trace data which may be pertinent to any category/filter of the trace criteria (Step 106). Trace output categories are defined which map to the filtering criteria, and each trace output entry which is produced during execution is associated with one or more of these categories. This data is stored in memory 10 for future use (Step 108). Accordingly, all of the information generated by the trace facility remains available to the user. These trace entries retain information regarding their origin, and thus remain usable, as opposed to the filtered text which is output by the prior art and may only be further filtered with regard to text strings found in the different entries, if at all.

> The trace entries may be stored in many forms, but it has been found that storing them as object oriented objects is preferable, since the nature of such objects lends itself to maintaining the identity or origin information of the data produced by the trace facility." (col. 4, lines 1-19)

Moreover, the Examiner argues that it would have been obvious to one of ordinary skill in the art at the time of the invention was made to modify the system disclosed by Smaha by filtering according to one or more attributes before analysis. Appellant respectfully disagrees with this assertion of "obviousness," especially in view of the ample evidence to the contrary.

For example, Smaha relates to an intrusion detection system, while Borchardt relates to a software debugger. To simply glean features from a software bugger, such as that of Borchardt, and combine the same with the *non-analogous art* of intrusion detection systems, such as that of Smaha, would simply be improper. Software debuggers detect code problems in newly created software, while an intrusion detection system detects hacker activity. "In order to rely on a reference as a basis for rejection of an appellant's invention, the reference must either be in the field of appellant's endeavor or, if not, then be reasonably pertinent to the particular problem with which the inventor was concerned." In re Oetiker, 977 F.2d 1443, 1446, 24 USPQ2d 1443, 1445 (Fed. Cir. 1992). See also In re Deminski, 796 F.2d 436, 230 USPQ 313 (Fed. Cir. 1986); In re Clay, 966 F.2d 656, 659, 23 USPQ2d 1058, 1060-61 (Fed. Cir. 1992) In view of the vastly different types of problems a software bugger addresses as opposed to an intrusion detection system, the Examiner's proposed combination is inappropriate.

In the previous action mailed March 22, 2004, it appears that the Examiner has generalized the description of Borchardt and Smaha to include "processing of

computer data." Appellant asserts that such generalization of the descriptions of the arts embodied by Borchardt and Smaha is improper, especially since appellant has clearly set forth the clear differences among the particular problems solved in the arts of intrusion detection systems and software buggers.

To simply generalize the arts associated with Borchardt and Smaha to the point that they are "analogous" in order to support the Examiner's obviousness argument, despite the fundamentally different problems which such arts attempt to solve, would frustrate the inventive concepts of appellant, especially in view of the manner in which Borchardt *teaches away* from the concepts of Smaha and would even *frustrate the purpose* thereof.

More importantly, the foregoing Borchardt excerpt and the remaining portions of the Borchardt reference, fail to disclose, teach or suggest any sort of "size of the audit log [being] reduced when the auditing system is run prior to the examination for detection of the patterns" (emphasis added). Instead, Borchardt simply discloses the detection of "inadvertent programming errors," or "bugs." This is vastly different from the "patterns" used for "intrusion detection purposes," as claimed by appellant.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable

expectation of success must both be found in the prior art and not based on appellant's disclosure. *In re Vaeck,* 947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

Appellant respectfully asserts that at least the first and third elements of the *prima facie* case of obviousness have not been met, for the reasons set forth hereinabove.

Group #2: Claims 3, 13, and 21

With respect to Claims 3, 13, and 21; the Examiner has relied upon U.S. Patent No. 4,713,754 to meet appellant's claimed "wherein the auditing system is configured to modify a system call jump table to cause at least one selected system call to execute code that causes the at least one target attribute to be recorded in response to detecting the at least one auditing criterion." The only mention of any sort of "table" in such reference is as follows:

> "Device Control Routines.
>
> In order to be able to create each A/LU 218 separately and to be able to load each A/LU 218 into the system dynamically, it is necessary to decouple all System Primitive Routines 404 and certain system functions, as described above, from the A/LUs 218. It is further preferable if, in doing so, the speed of execution of system primitive calls by the A/LUs 218 were enhanced. This is done, as shown in FIG. 4, by coding all System Primitive Routine 404 entry points as interrupt vectors through a Software Interrupt **Vector Table** 406 residing in a reserved area of Memory 106. Essentially, and as described below, all System Primitive Routine 404 calls by A/LUs 218, that is, by Tasks 112, are executed as software interrupt calls. As an initial step in such calls, the calling Task 112, operating through the stack handling facility of TM 304, pushes all necessary arguments onto its associated stack and, for some calls, puts its function code into a

designated register.

In addition to the above A/LU 218/Task 112 calls, certain system functions, such as system start and application program terminate, are called through software interrupts to maintain software version independency. The system also provides, as indicated in FIG. 4, an Entry **Address Table** 408 for other entry to System Primitive Routines 404.

Considering now the construction of a A/LU 218/Task 112 performing System Primitive Routine 404 calls through a software interrupt mechanism, the initial source code applications Modules 402 have associated with them a Link Module 410 which defines the relationships between system primitive calls and the entries in Software Interrupt Vector Table 406. Upon compilation and linking of the original source code Modules and Link Module 410, through Compiler and Linker 412, the modules are transformed into a single, linked, relocatable object File 416 wherein the original system primitive calls are linked into the appropriate interrupt vectors taken from Link Module 410." (emphasis added - col. 8, lines 1 - 40)

It appears that the Examiner has simply searched to find a disclosure of a "table," and then made an assertion that appellant's claim language is obvious in view of the resulting combination.

Appellant respectfully disagrees with this assertion, since the Examiner has not met appellant's claimed "auditing system [that] is configured to modify a system call jump table to cause at least one selected system call to execute code that causes the at least one target attribute to be recorded in response to detecting the at least one auditing criterion" (emphasis added). Only appellant teaches and claims an auditing system capable of modifying a system call jump table for the specific purpose of causing a selected system call to execute code that, in turn, causes the at least one target attribute to be recorded in response to detecting the at least one auditing criterion.

Since the references, when combined, must teach or suggest all the claim limitations (and the Examiner's proposed combination has failed to do so), appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met.

Group #3: Claims 7, 16, and 25

With respect to Claims 7, 16, and 25; the Examiner simply states that "the filtering of data from an audit log inherently reduces the amount of data stored in it," in order to reject appellant's claimed "producing the audit log comprises filtering the at least one target attribute to reduce an amount of data stored in the audit log."

In response, appellant points out that the Examiner's rejection fails, since the Examiner has not made a prior art showing of "producing the audit log comprises filtering the at least one target attribute." Thus, insofar as the patentability of such claim limitations is concerned, what is inherent in view of such claim limitations is irrelevant, since no prior art showing has been made.

Issue # 2

The Examiner has rejected Claims 4, 6, 13, 15, 22, and 24 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,557,742 to Smaha et al., in view of U.S. Patent No.: 5,513,317 to Borchardt et al., further in view of U.S. Patent No. 6,584,508 to Epstein et al., and further in view of Kernighan et al., "The UNIX Programming Environment," 1984.

Group #1: Claims 4, 13, and 22

With respect to Claims 4, 13, and 22, the Examiner admits that Epstein does not completely detail the attributes that are included in appellant's claimed system calls. By the Examiner's very admission, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the references, when combined, must teach or suggest all the claim limitations.

Group #2: Claims 6, 15, and 24

With respect to Claims 6, 15, and 24; the Examiner simply relies on a general Unix Programming Environment textbook to meet appellant's claimed "wherein the at least one auditing criterion includes ... an identifier for an application program from which the system call is being made." Such Unix Programming Environment textbook, however, fails to make any mention of appellant's claimed auditing criterion, let alone an auditing criterion that specifically includes an identifier for an application program from which the system call is being made, as claimed.

Appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the references, when combined, must teach or suggest all the claim limitations.

Issue # 3 - Group #1: Claims 5, 14, and 23

The Examiner has rejected Claims 5, 14, and 23 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,557,742 to Smaha et al., in view of U.S. Patent No.: 5,513,317 to Borchardt et al., and further in view of U.S. Patent No. 5,623,601 to Vu.

Specifically, the Examiner relies on the following excerpt from Vu to make a prior art showing of appellant's claimed: "wherein configuring the auditing system to record the at least one target attribute comprises:

compiling the audit specification to produce a kernel module;

loading the kernel module into a kernel of an operating system of the computer system; and

linking code from within the kernel module into system calls within the operating system."

> "an operating system executable by the gateway station, a kernel of the operating system having been modified so that the operating system:
>
> a) cannot forward any communications packet from the private network to the potentially hostile network or from the potentially hostile network to the private network; and
>
> b) will accept for processing any communications packet from either of the private network and the potentially hostile network provided that the packet is encapsulated with a hardware destination address that matches the device address of the gateway station on the respective networks; and" (col. 4, lines 51-64)
>
> "The gateway station in accordance with the invention will, however, detect a probe on any port and may be configured to set an alarm condition if repeated probes are attempted. The gateway station in accordance with the invention can also be configured to perform data sensitivity screening because all communications packets are delivered by the kernel to the application level where the data portion of each packet is passed from one in progress communications session to the other. Data sensitivity screening permits

> the detection of sophisticated intrusion techniques such as piggybacked protocols, and the like." (col. 6, lines 2-13)

Such excerpts, however, fail to disclose, teach or even suggest such claim limitations in the specific context of "configuring the auditing system to record the at least one target attribute," let alone "linking code from within the kernel module into system calls within the operating system."

Again, appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the references, when combined, must teach or suggest all the claim limitations.

In view of the remarks set forth hereinabove, all of the independent claims are deemed allowable, along with any claims depending therefrom.

IX APPENDIX OF CLAIMS (37 C.F.R. § 1.192(c)(9))

The text of the claims involved in the appeal is:

1. (Previously Amended) A method for providing content-based intrusion detection for a computer system by using an agile kernel-based auditing system, comprising:

receiving an audit specification;

wherein the audit specification specifies at least one target attribute to be recorded from a set of possible target attributes during an auditing process by the auditing system;

wherein the audit specification also specifies at least one auditing criterion that triggers recording of the at least one target attribute during the auditing process;

configuring the auditing system to record the at least one target attribute in response to detecting the at least one auditing criterion;

running the auditing system to produce an audit log by recording the at least one target attribute in response to detecting the at least one auditing criterion; and

examining the audit log to detect patterns for intrusion detection purposes;

wherein a size of the audit log is reduced when the auditing system is run prior to the examination for detection of the patterns.

2. (Original) The method of claim 1, further comprising:

detecting an event during the auditing process; and

in response to detecting the event, dynamically adjusting the auditing system during the auditing process to change the at least one auditing criterion and/or the at least one target attribute for subsequent operation of the auditing system.

3. (Original) The method of claim 1, wherein the auditing system is configured to modify a system call jump table to cause at least one selected system call to execute code that causes the at least one target attribute to be recorded in response to detecting the at least one auditing criterion.

4. (Previously Amended) The method of claim 1, wherein the at least one target attribute includes:

an argument from a system call;

a parameter of a process making the system call;

data read during the system call;

data written during the system call;

a parameter of a file involved in the system call; and

a parameter relating to a network communication involved in the system call.

5. (Previously Amended) The method of claim 1, wherein configuring the auditing system to record the at least one target attribute comprises:

compiling the audit specification to produce a kernel module;

loading the kernel module into a kernel of an operating system of the computer system; and

linking code from within the kernel module into system calls within the operating system.

6. (Previously Amended) The method of claim 1, wherein the at least one auditing criterion includes:

a user identifier for a process that is making a system call;

an identifier for an application program from which the system call is being made; and

an identifier for a file being accessed by the system call.

7. (Previously Amended) The method of claim 1, wherein producing the audit log comprises filtering the at least one target attribute to reduce an amount of data stored in the audit log.

8. (Previously Amended) The method of claim 1, wherein producing the audit log comprises:

determining at least one characteristic of the at least one target attribute; and

recording the at least one characteristic in the audit log.

9. (Original) The method of claim 1, wherein the audit specification is received from one of:

a user of the auditing system; and

an intrusion detection mechanism.

10. (Previously Amended) A computer-readable storage medium storing instructions that when executed by a computer cause the computer to

perform a method for providing content-based intrusion detection for a computer system by using an agile kernel-based auditing system, the method comprising:

receiving an audit specification;

wherein the audit specification specifies at least one target attribute to be recorded from a set of possible target attributes during an auditing process by the auditing system;

wherein the audit specification also specifies at least one auditing criterion that triggers recording of the at least one target attribute during the auditing process;

configuring the auditing system to record the at least one target attribute in response to detecting the at least one auditing criterion in response to detecting the at least one auditing criterion;

running the auditing system to produce an audit log by recording the at least one target attribute; and

examining the audit log to detect patterns for intrusion detection purposes;

wherein a size of the audit log is reduced when the auditing system is run prior to the examination for detection of the patterns.

11. (Previously Amended) The computer-readable storage medium of claim 10, wherein the method further comprises:

detecting an event during the auditing process; and

in response to detecting the event, dynamically adjusting the auditing system during the auditing process to change the at least one auditing criterion or the at least one target attribute for subsequent operation of the auditing system.

12. (Original) The computer-readable storage medium of claim 10, wherein the auditing system is configured to modify a system call jump table to

cause at least one selected system call to execute code that causes the at least one target attribute to be recorded in response to detecting the at least one auditing criterion.

13. (Previously Amended) The computer-readable storage medium of claim 10, wherein the at least one target attribute includes:

an argument from a system call;

a parameter of a process making the system call;

data read during the system call;

data written during the system call;

a parameter of a file involved in the system call; and

a parameter relating to a network communication involved in the system call.

14. (Previously Amended) The computer-readable storage medium of claim 10, wherein configuring the auditing system to record the at least one target attribute comprises:

compiling the audit specification to produce a kernel module;

loading the kernel module into a kernel of an operating system of the computer system; and

linking code from within the kernel module into system calls within the operating system.

15. (Previously Amended) The computer-readable storage medium of claim 10, wherein the at least one auditing criterion includes:

a user identifier for a process that is making a system call;

an identifier for an application program from which the system call is being made; and

an identifier for a file being accessed by the system call.

16. (Previously Amended) The computer-readable storage medium of claim 10, wherein producing the audit log comprises filtering the at least one target attribute to reduce an amount of data stored in the audit log.

17. (Previously Amended) The computer-readable storage medium of claim 10, wherein producing the audit log comprises:

determining at least one characteristic of the at least one target attribute; and

recording the at least one characteristic in the audit log.

18. (Original) The computer-readable storage medium of claim 10, wherein the audit specification is received from one of:

a user of the auditing system; and

an intrusion detection mechanism.

19. (Previously Amended) A apparatus for providing content-based intrusion detection for a computer system by using an agile kernel-based auditing mechanism, comprising:

an auditing mechanism that is configured to audit system calls;

a receiving mechanism that is configured to receive an audit specification;

wherein the audit specification specifies at least one target attribute to be recorded from a set of possible target attributes during an auditing process by the auditing mechanism;

wherein the audit specification also specifies at least one auditing criterion that triggers recording of the at least one target attribute during the auditing process;

an initialization mechanism that configures the auditing mechanism to record the at least one target attribute in response to detecting the at least one auditing criterion;

wherein the auditing mechanism is configured to produce an audit log by recording the at least one target attribute in response to detecting the at least one auditing criterion; and

an intrusion detection mechanism that is configured to examine the audit log to detect patterns for intrusion detection purposes;

wherein a size of the audit log is reduced when the auditing mechanism is run prior to the examination for detection of the patterns.

20. (Previously Amended) The apparatus of claim 19, wherein the initialization mechanism is further configured to:

detect an event during the auditing process; and

in response to detecting the event, to dynamically adjust the auditing mechanism during the auditing process to change the at least one auditing criterion or the at least one target attribute for subsequent operation of the auditing mechanism.

21. (Original) The apparatus of claim 19, wherein the auditing mechanism is configured to modify a system call jump table to cause at least one selected system call to execute code that causes the at least one target attribute to be recorded in response to detecting the at least one auditing criterion.

22. (Previously Amended) The apparatus of claim 19, wherein the at least one target attribute includes:

an argument from a system call;

a parameter of a process making the system call;

data read during the system call;

data written during the system call;

a parameter of a file involved in the system call; and

a parameter relating to a network communication involved in the system call.

23. (Original) The apparatus of claim 19, wherein the auditing mechanism is configured to:

compile the audit specification to produce a kernel module;

load the kernel module into a kernel of an operating system of the computer system; and to

link code from within the kernel module into system calls within the operating system.

24. (Previously Amended) The apparatus of claim 19, wherein the at least one auditing criterion includes:

a user identifier for a process that is making a system call;

an identifier for an application program from which the system call is being made; and

an identifier for a file being accessed by the system call.

25. (Original) The apparatus of claim 19, wherein the auditing mechanism is configured to produce the audit log by filtering the at least one target attribute to reduce an amount of data stored in the audit log.

26. (Previously Amended) The apparatus of claim 19, wherein the auditing mechanism is configured to produce the audit log by operations comprising:

determining at least one characteristic of the at least one target attribute; and

recording the at least one characteristic in the audit log.

27. (Original) The apparatus of claim 19, wherein the audit specification is received from one of:

a user of the auditing mechanism; and

the intrusion detection mechanism.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 971-2573. For payment of any additional fees due in connection with the filing of this paper, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1351 (Order No. NAI1P250/00.024.01).

Respectfully submitted,

By: _____ Date: 04/30/04

Kevin J. Zilka

Reg. No. 41,429

Silicon Valley IP Group, P.C.
P.O. Box 721120
San Jose, California 95172-1120
Telephone: (408) 971-2573
Facsimile: (408) 971-4660